or laptop devices, multiprocessor systems, microprocessor-based systems, set top boxes, programmable consumer electronics, mobile telephones, network PCs, minicomputers, mainframe computers, distributed computing environments that include any of the above systems or devices, and the like.

[0092] The invention may be described in the general context of computer-executable instructions, such as program modules, executed by one or more computers or other devices. Generally, program modules include, but are not limited to, routines, programs, objects, components, and data structures that perform particular tasks or implement particular abstract data types. The invention may also be practiced in distributed computing environments where tasks are performed by remote processing devices that are linked through a communications network. In a distributed computing environment, program modules may be located in both local and remote computer storage media including memory storage devices.

[0093] An interface in the context of a software architecture includes a software module, component, code portion, or other sequence of computer-executable instructions. The interface includes, for example, a first module accessing a second module to perform computing tasks on behalf of the first module. The first and second modules include, in one example, application programming interfaces (APIs) such as provided by operating systems, component object model (COM) interfaces (e.g., for peer-to-peer application communication), and extensible markup language metadata interchange format (XMI) interfaces (e.g., for communication between web services).

[0094] The interface may be a tightly coupled, synchronous implementation such as in Java 2 Platform Enterprise Edition (J2EE), COM, or distributed COM (DCOM) examples. Alternatively or in addition, the interface may be a loosely coupled, asynchronous implementation such as in a web service (e.g., using the simple object access protocol). In general, the interface includes any combination of the following characteristics: tightly coupled, loosely coupled, synchronous, and asynchronous. Further, the interface may conform to a standard protocol, a proprietary protocol, or any combination of standard and proprietary protocols.

[0095] The interfaces described herein may all be part of a single interface or may be implemented as separate interfaces or any combination therein. The interfaces may execute locally or remotely to provide functionality. Further, the interfaces may include additional or less functionality than illustrated or described herein.

[0096] In operation, computer **130** executes computer-executable instructions such as those illustrated in the figures to grant an application program access to a resource according to a privilege associated with the application program and with the resource. The systems and methods illustrated in the figures and described herein may be implemented in software or hardware or both using techniques some of which are well known in the art.

[0097] Manifest Example

[0098] The following examples further illustrate the invention. While some of the examples below include a reference to a registry, embodiments of the invention are not limited to a registry. Embodiments of the invention are operable with any mechanism for storing system settings. Attributes are inherited with some mechanisms, while inheritance is not a guarantee with other mechanisms. Table 1 below lists exemplary privileges in a manifest and describes the type of resource protection associated with each of the levels.

TABLE 1

Exemplary Privileges.

Privileges	Type of Protection
readOnlyIgnoreWrites	Read only - Files or settings associated with this privilege are modifiable only by the operating system when installing or servicing (e.g., upgrading). Other attempts to write to this file or setting are silently ignored (e.g., return a success response even though no write happens).
readOnlyFailWrites	Read only - Files or settings associated with this privilege are modifiable only by the operating system when installing or servicing (e.g., upgrading). Other attempts to write to this file or setting are explicitly ignored (e.g., return a failure response).
OSOnlyIgnoreWrites	Files or settings associated with this privilege are modifiable only by an operating system component. Other attempts to write to this file or setting are silently ignored (e.g., return a success response even though no write happens).
OSOnlyFailWrites	Files or settings associated with this privilege are modifiable only by an operating system component. Other attempts to write to this file or setting are explicitly ignored (e.g., return a failure response).
change recording	Different values stored for settings associated with this privilege will be retained on a per-application basis but visible on a global basis (last application which wrote). The current global value is rolled